

FFFFFFF	000000000	RRRRRRRRRRR	RRRRRRRRRRR	TTTTTTTTTTTTT	LLL
FFFFF	000000000	RRRRRRRRRRR	RRRRRRRRRRR	TTTTTTTTTTTTT	LLL
FFFFF	000000000	RRRRRRRRRRR	RRRRRRRRRRR	TTTTTTTTTTTTT	LLL
FFF	000	000	RRR	RRR	TTT
FFF	000	000	RRR	RRR	TTT
FFF	000	000	RRR	RRR	TTT
FFF	000	000	RRR	RRR	TTT
FFF	000	000	RRR	RRR	TTT
FFF	000	000	RRR	RRR	TTT
FFF	000	000	RRR	RRR	TTT
FFF	000	000	RRR	RRR	TTT
FFF	000	000	RRR	RRR	TTT
FFF	000	000	RRR	RRR	TTT
FFF	000	000	RRR	RRR	TTT
FFF	000	000	RRR	RRR	TTT
FFF	000	000	RRR	RRR	TTT
FFF	000	000	RRR	RRR	TTT
FFF	000	000	RRR	RRR	TTT
FFF	000	000	RRR	RRR	TTT
FFF	000	000	RRR	RRR	TTT
FFF	000	000	RRR	RRR	TTT
FFF	000	000	RRR	RRR	TTT
FFF	000	000	RRR	RRR	TTT
FFF	000	000	RRR	RRR	TTT
FFF	000	000	RRR	RRR	TTT
FFF	000000000	RRR	RRR	RRR	LLLLLLLLLLLL
FFF	000000000	RRR	RRR	RRR	LLLLLLLLLLLL
FFF	000000000	RRR	RRR	RRR	LLLLLLLLLLLL

FILEID**FORREWIND

E 13

FOR
1-C

FFFFFFFFF	000000	RRRRRRR	RRRRRRR	EEEEEEEEE	WW	WW	IIIIII	NN	NN	DDDDDDDD		
FFFFFFFFF	000000	RRRRRRR	RRRRRRR	EEEEEEEEE	WW	WW	IIIIII	NN	NN	DDDDDDDD		
FF	00	00	RR	RR	EE	WW	WW	II	NN	NN	DD	DD
FF	00	00	RR	RR	EE	WW	WW	II	NN	NN	DD	DD
FF	00	00	RR	RR	EE	WW	WW	II	NNNN	NN	DD	DD
FF	00	00	RR	RR	EE	WW	WW	II	NNNN	NN	DD	DD
FF	00	00	RRRRRRR	RRRRRRR	EEEEEEEEE	WW	WW	II	NN	NN	DD	DD
FF	00	00	RRRRRRR	RRRRRRR	EEEEEEEEE	WW	WW	II	NN	NN	DD	DD
FF	00	00	RR RR	RR RR	EE	WW	WW	II	NN	NNNN	DD	DD
FF	00	00	RR RR	RR RR	EE	WW	WW	II	NN	NNNN	DD	DD
FF	00	00	RR RR	RR RR	EE	WWWW	WWWW	II	NN	NN	DD	DD
FF	00	00	RR RR	RR RR	EE	WWWW	WWWW	II	NN	NN	DD	DD
FF	000000	RR	RR	RR	RR	WW	WW	IIIIII	NN	NN	DDDDDDDD
FF	000000	RR	RR	RR	RR	WW	WW	IIIIII	NN	NN	DDDDDDDD

LL	IIIIII	SSSSSSS
LL	IIIIII	SSSSSSS
LL	II	SS
LL	II	SS
LL	II	SS
LL	II	SSSSSS
LL	II	SSSSSS
LL	II	SS
LL	II	SS
LL	II	SS
LLLLLLLLL	IIIIII	SSSSSSS
LLLLLLLLL	IIIIII	SSSSSSS

```
: 1      0 MODULE FOR$REWIND ( ! FORTRAN REWIND Statement
: 2      0 IDENT = '1-007' ! File: FORREWIND.B32 ! Edit SBL1007
: 3      0 ) =
: 4      1 BEGIN
: 5      1
: 6      1 ****
: 7      1 *
: 8      1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
: 9      1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
:10      1 * ALL RIGHTS RESERVED.
:11      1 *
:12      1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
:13      1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
:14      1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
:15      1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
:16      1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
:17      1 * TRANSFERRED.
:18      1 *
:19      1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
:20      1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
:21      1 * CORPORATION.
:22      1 *
:23      1 *
:24      1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
:25      1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
:26      1 *
:27      1 *
:28      1 ****
:29      1 *
:30      1 *
:31      1 ++
:32      1 * FACILITY: FORTRAN Support Library, user callable
:33      1 *
:34      1 * ABSTRACT:
:35      1 *
:36      1 * Contains routine FOR$REWIND: rewind a FORTRAN sequential
:37      1 * access file.
:38      1 *
:39      1 * ENVIRONMENT: Mixture of AST level or not.
:40      1 *
:41      1 * AUTHOR: Jonathan M. Taylor, CREATION DATE: 10-OCT-77
:42      1 *
:43      1 * MODIFIED BY:
:44      1 *
:45      1 * Jonathan M. Taylor, 10-OCT-77 : VERSION 0
:46      1 * Previous edit history removed. SBL 16-June-1982
:47      1 * 1-001 - Update version number and copyright notice. JBS 16-NOV-78
:48      1 * 1-002 - Change REQUIRE file names from FOR... to OTS... JBS 06-DEC-78
:49      1 * 1-003 - Change prefix of LUN literals from OPEN to LUB. JBS 13-DEC-78
:50      1 * 1-004 - Implement ERR= and IOSTAT=. SBL 1-May-1979
:51      1 * 1-005 - Error instead of no-op on not open or direct. SBL 2-May-1979
:52      1 * 1-006 - 1-005 is a mistake. No-op if not open, error if not
:53      1 * sequential org and access. SBL 16-May-1979
:54      1 * 1-007 - Allow errors RMSS_IOP, RMSS_BOF and RMSS_EOF from $REWIND.
:55      1 * Move declaration of ACTUALCOUNT. Add SWITCHES. SBL 16-June-1982
:56      1 --
```

```
: 58      0057 1 : SWITCHES:  
59      0058 1 :  
60      0059 1 :  
61      0060 1 :  
62      0061 1 : SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);  
63      0062 1 :  
64      0063 1 :  
65      0064 1 : LINKAGES:  
66      0065 1 :  
67      0066 1 : REQUIRE 'RTLIN:OTSLNK';           ! Define all linkages  
68      0495 1 :  
69      0496 1 :  
70      0497 1 : TABLE OF CONTENTS:  
71      0498 1 :  
72      0499 1 :  
73      0500 1 : FORWARD ROUTINE  
74      0501 1 :          FOR$REWIND;           ! FORTRAN REWIND statement processor  
75      0502 1 :  
76      0503 1 :  
77      0504 1 : INCLUDE FILES:  
78      0505 1 :  
79      0506 1 :  
80      0507 1 : REQUIRE 'RTLML:FORERR';           ! FORTRAN error number definitions  
81      0575 1 : REQUIRE 'RTLML:OTSLUB';           Logical Unit Block definitions  
82      0715 1 : REQUIRE 'RTLIN:OTSMAC';           Macros  
83      0909 1 : REQUIRE 'RTLIN:RTLPSECT';          Define DECLARE_PSECTS macro  
84      1004 1 : REQUIRE 'RTLML:OTSISB';           ISB definitions  
85      1172 1 : REQUIRE 'RTLML:FORPAR';           FORTRAN inter-module parameters  
86      1195 1 :          LIBRARY 'RTLSTARLE';          ! STARLET library for macros and symbols  
87      1196 1 :  
88      1197 1 : MACROS:  
89      1198 1 :          NONE  
90      1199 1 :  
91      1200 1 :  
92      1201 1 : EQUATED SYMBOLS:  
93      1202 1 :          NONE  
94      1203 1 :  
95      1204 1 :  
96      1205 1 :  
97      1206 1 : OWN STORAGE:  
98      1207 1 :          NONE  
99      1208 1 :  
100     1209 1 :  
101     1210 1 : EXTERNAL REFERENCES:  
102     1211 1 :  
103     1212 1 : EXTERNAL ROUTINE  
104     1213 1 :          FOR$SIOSTAT_HND;           ! error condition handler  
105     1214 1 :          FOR$$SIGNAL_STO: NOVALUE;       convert error number and signal  
106     1215 1 :          FOR$$CB_PUSH: JSB_CB_PUSH_NOVALUE;    ! create LUB/ISB/RAB, if needed  
107     1216 1 :          FOR$$CB_POP: JSB_CB_POP_NOVALUE;     ! return I/O system to previous state  
108     1217 1 :  
109     1218 1 :  
110     1219 1 : PSECT DECLARATIONS:  
111     1220 1 :  
112     1221 1 :  
113     1222 1 :          DECLARE_PSECTS (FOR);        ! declare PSECTS for FOR$ facility
```

```
: 115      1223 1 GLOBAL ROUTINE FOR$REWIND (
: 116          1224 1     UNIT
: 117          1225 1     ERR_EQL)
: 118          1226 1     =
: 119          1227 1
: 120          1228 1     ++
: 121          1229 1     FUNCTIONAL DESCRIPTION:
: 122          1230 1
: 123          1231 1     Perform RMS rewind operation on the file specified by the
: 124          1232 1     UNIT parameter.
: 125          1233 1
: 126          1234 1     FORMAL PARAMETERS:
: 127          1235 1
: 128          1236 1     UNIT.rlu.v           Logical unit number
: 129          1237 1     ERR_EQL.rl.v        If 0 or not present, signal errors
: 130          1238 1           If non-zero, unwind to caller.
: 131          1239 1
: 132          1240 1     IMPLICIT INPUTS:
: 133          1241 1
: 134          1242 1     LUB$V_DIRECT        This unit has previously been specified
: 135          1243 1           for direct access by an OPEN statement or
: 136          1244 1           DEFINE FILE.
: 137          1245 1     LUB$V_OPENED        This unit has already been opened by
: 138          1246 1           an OPEN statement or default open.
: 139          1247 1
: 140          1248 1     IMPLICIT OUTPUTS:
: 141          1249 1
: 142          1250 1     LUB$L_LOG_RECNO    set to 1.
: 143          1251 1
: 144          1252 1     ROUTINE VALUE:
: 145          1253 1
: 146          1254 1     An IOSTAT value.
: 147          1255 1
: 148          1256 1     SIDE EFFECTS:
: 149          1257 1
: 150          1258 1     SIGNAL_STOPS FOR$REWERR if a non-EOF error is returned from
: 151          1259 1           the RMS rewind call.
: 152          1260 1
: 153          1261 1     !!--
: 154          1262 1
: 155          1263 2     BEGIN
: 156          1264 2
: 157          1265 2     GLOBAL REGISTER
: 158          1266 2     CCB = 11: REF_BLOCK[, BYTE];
: 159          1267 2
: 160          1268 2     LOCAL
: 161          1269 2     STATUS,           ! Return status from $REWIND
: 162          1270 2     L_UNWIND_ACTION: VOLATILE, ! Unwind action code (FOR$K_INWIND{POP or NOP})
: 163          1271 2     L_ERR_EQL_PRES: VOLATILE; ! 1 if ERR= present
: 164          1272 2
: 165          1273 2     BUILTIN
: 166          1274 2     ACTUALCOUNT;
: 167          1275 2
: 168          1276 2     ENABLE
: 169          1277 2     FOR$IOSTAT_HND (L_UNWIND_ACTION, L_ERR_EQL_PRES); ! pass info to error handler
: 170          1278 2
: 171          1279 2
```

```
172      1280 2      !+ Determine if ERR= is present.  
173      1281 2      !-  
174      1282 2  
175      1283 2  
176      1284 2      IF ACTUALCOUNT () GTR 1  
177      1285 2      THEN L_ERR_EQL_PRES = .ERR_EQL  
178      1286 2      ELSE L_ERR_EQL_PRES = 0;  
179      1287 2  
180      1288 2  
181      1289 2  
182      1290 2      !+ Set up error handler conditions in case CB_PUSH bombs  
183      1291 2      !-  
184      1292 2  
185      1293 2  
186      1294 2      L_UNWIND_ACTION = FORSK_UNWINDNOP;  
187      1295 2  
188      1296 2      !+ Get a LUB for this logical unit.  
189      1297 2      ! On return, CCB points to the current control block.  
190      1298 2      !-  
191      1299 2  
192      1300 2  
193      1301 2      FOR$SCB_PUSH (.UNIT, LUB$K_LUN_MIN);  
194      1302 2  
195      1303 2      !+ Unwind action (if an error occurs) is now to pop a LUB.  
196      1304 2      !-  
197      1305 2  
198      1306 2  
199      1307 2      L_UNWIND_ACTION = FORSK_UNWINDPOP;  
200      1308 2  
201      1309 2      !+ Check the LUB. If file is not open, then this is a no-op.  
202      1310 2      ! Else must be sequential organization and access.  
203      1311 2      !-  
204      1312 2  
205      1313 2  
206      1314 2      IF .CCB [LUB$V_OPENED]  
207      1315 2      THEN IF NOT .CCB [LUB$V_DIRECT] AND NOT .CCB [LUB$V_NOTSEQORG]  
208      1316 2      THEN BEGIN  
209      1317 2      THEN BEGIN  
210      1318 3      !+ Call RMS to REWIND the file, all failure codes returned  
211      1319 3      ! cause a SIGNAL_STOP to occur, except for IOP, EOF or BOF.  
212      1320 3      !-  
213      1321 3  
214      1322 3  
215      1323 3  
216      1324 3  
217      1325 4      IF NOT (STATUS = $REWIND (RAB = .CCB))  
218      1326 3      THEN BEGIN  
219      1327 4      IF .STATUS NEQ RMSS_IOP AND  
220      1328 4      .STATUS NEQ RMSS_EOF AND  
221      1329 4      .STATUS NEQ RMSS_BOF  
222      1330 4      THEN BEGIN  
223      1331 4      BEGIN  
224      1332 5      FOR$$SIGNAL_STOP (FORSK_REWERR);  
225      1333 5      RETURN 0;  
226      1334 5      END;  
227      1335 4      END;  
228      1336 3
```

```

229      1337
230      1338
231      1339
232      1340
233      1341
234      1342
235      1343
236      1344
237      1345
238      1346
239      1347
240      1348
241      1349
242      1350
243      1351
244      1352
245      1353
246      1354
247      1355
248      1356
249      1357
250      1358
251      1359
252      1360
253      1361
254      1362
255      1363

;+ Clear APPEND flag - OK for backspace
;- CCB[LUB$V_APPEND] = 0;

;+ Set the logical record number to 1.
;- CCB[LUB$L_LOG_RECNO] = 1;

        END
ELSE
BEGIN
FOR$SIGNAL_STO (FOR$K_REWERR);
RETURN 0;
END;

;+ Return the file system to its former state.
;- FOR$SCB_POP ();
RETURN 0;           ! Success IOSTAT value
END;

```

```

.TITLE FOR$REWIND
.IDENT \1-007\

.EXTRN FOR$SIOSTAT_HND
.EXTRN FOR$SIGNAL_STO
.EXTRN FOR$SCB_PUSH, FOR$SCB_POP
.EXTRN SYSSREWIND

.PSECT _FOR$CODE,NOWRT, SHR, PIC,2

.ENTRY FOR$REWIND, Save R2,R11 : 1223
SUBL2 #4, SP
CLRL L_ERR_EQL_PRES : 1263
CLRL L_UNWIND_ACTION
MOVAL 78, (FP)
CMPB (AP), #1 : 1284
BLEQU 1S
MOVL ERR_EQL, L_ERR_EQL_PRES : 1286
BRB 2S
CLRL L_ERR_EQL_PRES : 1288
MOVL #T, L_UNWIND_ACTION : 1294
CLRL R0 : 1301
MOVL UNIT, R2
JSB FOR$SCB_PUSH
CLRL L_UNWIND_ACTION : 1307
BLBC -4((CB), -5$ : 1314
BBS #4, -4((CB), 4$ : 1316
BBS #3, -95((CB), 4$ : 1317

```

00000000G	00	5B	DD 0003D	PUSHL	CCB		1325
		01	FB 0003F	CALLS	#1, SY\$REWIND		
00018574	1B	50	E8 00046	BLBS	STATUS, 3\$		
	BF	50	D1 00049	CMPL	STATUS, #99700		1328
0001827A	8F	12	13 00050	BEQL	3\$		
		50	D1 00052	CMPL	STATUS, #98938		1329
00018198	8F	09	13 00059	BEQL	3\$		
		50	D1 0005B	CMPL	STATUS, #98712		1330
		0A	12 00062	BNEQ	4\$		
FD	AB	20	8A 00064	3\$: BICB2	#32, -3(CC(B))		1342
EO	AB	01	D0 00068	MOVL	#1, -32(CC(B))		1348
		0B	11 0006C	BRB	5\$		1316
00000000G	00	14	DD 0006E	4\$: PUSHL	#20		1353
		01	FB 00070	CALLS	#1, FOR\$\$SIGNAL_STO		
		06	11 00077	BRB	6\$		
00000000G	00	00	16 00079	5\$: JSB	FOR\$\$CB_POP		1354
		50	D4 0007F	6\$: CLRL	R0		1361
			04 00081	RET			1363
			0000 00082	7\$: .WORD	Save nothing		1263
50		08	AC D0 00084	MOVL	8(AP), R0		
50		04	AO D0 00088	MOVL	4(R0), R0		
		F8	AO 9F 0008C	PUSHAB	L_ERR EQL PRES		
		FC	AO 9F 0008F	PUSHAB	L_UNWIND_ACTION		
		02	DD 00092	PUSHL	#2		
		5E	DD 00094	PUSHL	SP		
00000000G	7E	04	AC 7D 00096	MOVQ	4(AP), -(SP)		
		03	FB 0009A	CALLS	#3, FOR\$SIOSTAT_HND		
		04	000A1	RET			

; Routine Size: 162 bytes, Routine Base: _FOR\$CODE + 0000

```
; 256      1364 1
; 257      1365 1 END
; 258      1366 0 ELUDOM           !End of module
```

PSECT SUMMARY

Name	Bytes	Attributes
_FOR\$CODE	162	NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2)

Library Statistics

File	----- Symbols -----	Pages Mapped	Processing Time
	Total Loaded Percent		
\$_255\$DUA28:[SYSLIB]STARLET.L32;1	9776 7 0	581	00:01.3

COMMAND QUALIFIERS

```
: BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACE/LIS=LISS:FORREWIND/OBJ=OBJ$:FORREWIND MSRC$:FORREWIND/UPDATE=(ENH$:FORREWIND  
: )
```

```
: Size:        162 code + 0 data bytes  
: Run Time:    00:11.5  
: Elapsed Time: 00:33.2  
: Lines/CPU Min: 7158  
: Lexemes/CPU-Min: 40302  
: Memory Used: 146 pages  
: Compilation Complete
```

0183 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

FORREADF
LIS

FORREADO
LIS

FORRECPO
LIS

FORREWSU
LIS

FORREADK
LIS

FORREWIND
LIS

FORSIGNAL
LIS

FORREADSF
LIS

FORREWSO
LIS

FORREADSN
LIS

FORSECONDS
LIS

FORREADDU
LIS

FORREADSU
LIS

FORREADIL
LIS

FORREADKF
LIS

FORREWSF
LIS

FORREADKU
LIS

FORREADSL
LIS FORREADSO
LIS